(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 450 521 B1

(12) EUROPEAN PATENT SPECIFICATION

(45) Date of publication and mention of the grant of the patent:
05.03.1997 Bulletin 1997/10

(51) Int Cl.^{6}: **G06F 15/80**

(21) Application number: **91105033.4**

(22) Date of filing: **28.03.1991**

(54) **A learning method for a data processing system**

Lernverfahren für Datenverarbeitungsanlage

Méthode d'apprentissage pour un système de traitement de données

(84) Designated Contracting States:
BE DE FR GB IT NL SE

(30) Priority: **04.04.1990 JP 88162/90**

(43) Date of publication of application:
09.10.1991 Bulletin 1991/41

(73) Proprietors:
- **YOZAN INC.**
  Tokyo 155 (JP)
- **SHARP KABUSHIKI KAISHA**
  Osaka 545 (JP)

(72) Inventors:
- **Takatori, Sunao**
  Tokyo 155 (JP)
- **Yamamoto, Makoto**
  Tokyo 155 (JP)

(74) Representative: **Grünecker, Kinkeldey,
Stockmair & Schwanhäusser Anwaltssozietät
Maximilianstrasse 58
80538 München (DE)**

(56) References cited:
- **IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS vol. 1, 24 July 1988, SAN DIEGO,USA pages 465 - 472 SHEPANSKI 'Fast learning in artificial neural systems : Multilayer perceptron training using optimal estimation'**
- **IEEE FIRST INTERNATIONAL CONFERENCE ON NEURAL NETWORKS vol. 4, 21 June 1987, SAN DIEGO,USA pages 635 - 642 MCAULAY 'Engineering design neural networks using split inversion learning'**
- **AIP CONFERENCE PROCEEDINGS 151 : NEURAL NETWORKS FOR COMPUTING 1986, SNOWBIRD, USA pages 283 - 298 LAPEDES 'Programming a massively parallel computation universal system : static behaviour'**

Printed by Jouve, 75001 PARIS (FR)

Description

FIELD OF THE INVENTION

The present invention relates to training method applied to a data processing system according to the concept of a neural network.

BACKGROUND OF THE INVENTION

The neural network according to a data processing system of this type is structured with layers by preparing neuron model (hereinafter called "neuron") 1, shown in Fig. 3, in parallel. Neurons in each layer is synapse connected to all neurons in other adjacent layers, so as to input and output data. According to neuron 1, data 0 is output in accordance with the comparison result between the sum of multiplied input data from outside I1, I2, I3...In by weights W1, W2, W3...Wn and threshold θ.

Various comparison methods are possible. For example, when normalization function 1[f] is applied, output data 0 is express as follows:

$$0=1[\Sigma Wn \cdot In-\theta] \quad (1)$$

That is; when ΣWn In exceed threshold θ, the neuron ignites so that output data 0 becomes "1"; and when ΣWn·In is smaller than threshold θ, output data becomes "0".

A conventional neural network is structured to form neural layers by preparing above neurons in parallel and to connect the above neural layers in series. Neural layer, for example, comprises 3 layers; input layer, middle layer and output layer, as it is described by Perceptrons proposed by Rosenblatt, in which neurons of each layer are synapse connected to all neurons of other adjacent layer.

SUMMARY OF THE INVENTION

According to the above data processing system, the operation of weight optimization of synapse of each neuron is called "learning". The improvement of the efficiency and the guarantee of the realization of above are the important subjects to be investigated. For example, as to a back propagation method, which came to public notice recently, the problems of the escapement from local minimum and a convergent time are left. The above tendency is large especially with respect to the neural network including a plurality of middle layers.

The present invention is invented so as to solve the above problems of the prior art and has an object to provide a learning method performable of an efficient learning according to the middle layers.

A learning method according to the present invention is characterized as set out in claim 1.

It is essential to increase middle layers for many data processings. As the calculation methods for the nearest approximation, there are the minimum square approximation, etc.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 shows a rough structural diagram of a character recognition system according to an embodiment of the present invention.

Fig. 2 shows a schematic diagram of neurons in an input layer, a middle layer and an output layer.

Fig. 3 shows a schematic diagram of a neuron.

Fig. 4 is a graph showing the relationship between the number of learnings and weight change.

1 and N...neuron, 31...input layer, 32...middle layer, 33...output layer.

PREFERRED EMBODIMENT OF THE PRESENT INVENTION

Hereinafter, the present invention is described according to the embodiments with referring to the attached drawings.

Fig. 1 shows a character recognition system comprising a data processing system according to a learning method of an embodiment of the present invention. This character recognition system comprises: a video camera 10, a preprocessing system 20, a data processing system 30, a post-processing system 40 and display 50. A video camera 10 is prepared for inputting characters and is connected to a preprocessing system 20. A preprocessing system 20 is, for example, an image processing system, is generally and conventionally known, that extracts the characteristics data of input character (for example, number of end points, number of branch points, etc.) and outputs this characteristics data to a data processing system 30. A data processing system 30 structures the neural network, as it is described later, and outputs data to a post-processing system 40 according to the recognition result by recognizing an input character from a preprocessing system 20 according to its characteristics data. Recognition sign is, for example, a character code. A post-processing system 40 stores the above output data as, for example, a data of a work processor, and outputs it to a display 20, simultaneously. Display 50, for example, of the CRT type, displays recognized character by a data processing system 30 onto an image.

The neural network structuring a data processing system 30 is structured as a part of hardware. This data processing system 30, typically indicated in Fig. 1, comprised input layer 31, middle layer 32 and output layer 33 in which middle layer 32 is arranged between input layer 31 and output layer 33, as it is understood from this figure. According to the present embodiment, each layer 31, 32 and 33 comprises an equivalent number of neurons N, in which neurons N of input layer 31 are connected to all neurons N of middle layer 32, and neurons

N of middle neural layer 32 is connected to all neurons N of output layer 33.

Each neuron N outputs either "1" or "0" according to the formula (1) of the normalization function, as it is described with referring to Fig. 3. Neuron N is implemented, for example, by operation amplifiers. Weight Wn to be multiplied to a data input to each neuron N is obtained, for example, by a variable resistance connected to an input terminal of an operation amplifier. Threshold function is realized by, for example, switching elements. Accordingly, weight Wn is changed and output data is outputted by changing variable resistance with respect to the output data of each neuron, so as to perform the learning control.

Fig. 2 shows typical input layer 31, middle layer 32 and output layer 33, comprising equivalent number of neurons among themselves. Here, it is determined that number of neurons in each layer 31, 32 and 33 is 36, for the simple explanation, in which 6 neurons are arranged in horizontal direction and 6 neurons are arranged in vertical direction. When the location of a neuron at the down left corner in this figure is determined as an origin, a neuron at the i-th and j-th location from the left and from the bottom, respectively, is determined as Nij.

Each neuron of an input layer 31 ignites with respect to characteristics data of character input through a video camera 10. For example, it is deemed that number of end points is expressed by ignition pattern of N11, N12, N21 and N22; and number of branch points is expressed by ignition patterns of N13, N14, N23 and N24. Accordingly, ignition pattern of neurons of an input layer 31 is determined in accordance with an input character, artificially.

On the other hand, at on output layer 33, an character is expressed, for example, by neuron Nii on a diagonal line connecting neurons at down left corner and down right corner. Accordingly, neurons on the above diagonal line express character codes of a character. The ignition pattern of neurons of an output layer 33 is determined artificially. According to the present embodiment, the number of ignition patterns of neuron Nii on the diagonal line is 64. Therefore, it is possible to recognize 64 kinds of characters, according to the present embodiment, so that recognition of, for example, the alphabet is possible.

Before learning of character recognition is performed, neurons of output layer 33 do not ignite even if character data in input to a data processing system 30. The ignition of a neuron makes learning possible, and learning will be completed when a predetermined ignition pattern is output in accordance with input character data. The input and output patterns for learning is the representative data to be processed at the neural network. Data to be processed actually covers a wide area. Learning is continuously executed until a suitable association is performed with respect to the above representative input and output data. Therefore, when learning is completed, input layer 31 and output layer 33 output artificially determined ignition pattern, as it is described above, according to input character data. Then it is assumed that the ignition patterns of input layer 31, middle layer 32 and output layer 33 are changed smoothly over the above layers 31, 32 and 33. According to the present embodiment, weights of middle layer 32 and output layer 33 are changed on a process of learning so that the ignition patterns of the above layer change smoothly.

Here, the location of a neuron at down left corner is determined as on origin, and an output data of neuron Nij of an input layer 31 at the i-th location from left and from down, respectively, is determined as aij. Similar to the above, output data of neuron Nij of a middle layer 32 is determined as bij; and output data of neuron Nij of an output layer 33 is determined as cij.

The sum of squares of the differences among each neuron of each layer 31, 32 and 33 is expressed as follows:

$$f=\Sigma\{(aij-bij)^2+(bij-cij)^2\}$$

Accordingly, the above sum of squares f is the square distance with respect to the relationship between ignition patterns. Here, Σ indicates that i and j take the sum from 1 to 6.

Once an input character is determined, aij and cij are fixed, but not bij. Therefore, bij, when the sum of squares f is the minimal value, is easily calculated by, for example, a method of the least squares as is generally know. According to the learning of character recognition, weights of each neuron of a middle layer 32 and a output layer 33 are increased by predetermined value so as to obtain bij, i.e. ignition pattern, obtained as above. Hereinafter, the above processing is described with referring Fig. 3. Weights of synapses (in this case, "W2" and "3") is, for example, increased by 5% with respect to input data from ignited neuron (for example, "I2" and "I3") among neurons of input layer connected with neuron 1 when output data O(bij) of neuron 1 of a middle layer is the value at the ignition (for example "1"). According to weights of synapse with respect to the neurons of output layer, it is processed as similar to the above. As it is described above, weights of synapse are, for example, increased by 5% with respect to neurons of a middle layer that is determined to be ignited by a method of least squares.

Accordingly, weights of a middle layer 32 and an output layer 33 are increased so that the ignition patterns are smoothly changed over layers of an input layer 31, a middle layer 32 and an output layer 33. Here, when each neuron of an input layer 31 and output layer 33 is determined to ignite with closer frequency as possible with respect to all input character, the equivalent ignition among each neuron of a middle layer 32 becomes possible. Accordingly, it is possible to prevent falling into a local minimum. Furthermore, each neuron of a middle layer 31 will ignite approximately equivalently. There-

fore, evasion of the occurrence of neurons, which do not ignite, is possible so as to function the neurons of a middle layer 32 efficiently.

The weight increase of synapse for a learning change as shown in Fig. 4 with respect to the number of learnings. The learning for a whole system is gradually performed by a plurality of learnings, and a slight adjustment is performed with a little change at the end of learning. Learning speed is heightened as weight increases rapidly at the beginning of learning.

According to the above embodiment, there are 3 layers; a middle layer 32 are connected to an input layer 31 and an output layer 33. However, the present invention can be applied not only to the neural network with 3 layers but also the neural network with more then 3 layers. In this case, one middle layer should be selected first, then weights of synapses of neurons at the middle layer and an output layer are to be increased so that the sum of squares of the ignition patterns becomes the minimum value between that of in a middle layer and in an input layer 31 and output layer 33. After completing the learning with respect to all input characters up to the predetermined level, the next learning is performed by increasing the weights of synapses of neurons in the second middle layer and an output layer so that the sum of squares of the ignition pattern of the second middle layer becomes the minimum according to the ignition patterns of the other 2 layers adjacent to the above middle layer (the first middle layer and an input layer or an output layer). Accordingly, weights distribution with respect to the neural network with 4 layers is obtained. The same processing manner is applied to the neural network with more than 4 layers.

The method for deciding weights distribution of middle layer 32 and output layer 33 can be conceivable various method other than the first embodiment above.

For example, the absolute value of the difference of output data of neuron of each layer can be considered. That is, "f" of Hamming distance is considered, which is the accumulated value of laij-bijl and lbij-cijl regarding all neurons.

$$f=\Sigma\{|aij-bij|+|bij+cij|\}$$

Here, laij-bijl is the absolute value of the difference between an output data of a neuron of input layer 32 and an output data of a neuron of middle layer 32, and lbij+cijl is the absolute value of the difference between an output data of middle layer 32 and an output data of output layer 33. In this embodiment, "." is the accumulation from 1 to 6 with respect to "i" and "j".

"aij" and "cij" are fixed when input character is decided, but only "bij" is not fixed. Therefore, "bij" on the minimum value of Hamming distance "f" can be easily found out by general method. On learning of character recognition, the weights of each neuron of middle layer 32 and output layer 33 are increased by predetermined value in order to obtain "bij" of ignition pattern.

This is explained below referring Fig. 3. When output data "O(bij)" of the middle layer is the ignition value ("1", for example), synapse weight ("W2" and "3") corresponding to input data from ignited neuron ("I2" and "I3", for example) among the neurons of input layer connected to neuron 1 is increased 5%, for example. The weight of synapse of a neuron of output layer is processed in the same way, therefore, the weight of synapse corresponding to the neuron of the middle layer decided to be ignited from Hamming distance is increased 5%, for example.

The same effect of the first embodiment can be obtained by the second embodiment, too.

The second embodiment can be also applied to a neural network with neural layers of 4 or more, other than the neural network with 3 layers, similarly to the first embodiment. In this case, selecting the first middle layer, the weight of the synapses in the first middle layer and output layer are increased in order for the sum to be the minimum value, that is, in order for Hamming distance to be the minimum value: here, the "sum means the total value of the absolute value of the difference between the output data of each neuron of input layer 31 and that of the first middle layer, and the absolute value of the difference between the output data of each neuron of the first middle layer and that of output layer 33. When the learning up to the predetermined stage is completed with respect to all input characters, the second middle layer is newly added and the weight of the synapse is increased in the second middle layer and the layer connected to the output side of the second middle layer (output layer or the first middle layer), in the same way of the first middle layer in order for Hamming distance to be the minimum value. Consequently, weight distribution for 4 layers can be obtained. The same processing is executed for the case with 5b layers or more.

The third embodiment of the present invention is described next.

The coordination of neuron Nij in each layer 31, 32 and 33 is the same and they are corresponding to one another.

In order for ignition pattern of each layer of 31, 32 and 33 to change smoothly, the neuron of middle layer 32 has the tendency to ignite when corresponding neuron in input layer 31 and output layer 33 both ignite. Therefore, in this embodiment, the weight of synapse of predetermined neuron of middle layer 32 and output layer 33 is increased by the predetermined rate in order for each neuron in middle layer 32 ignite easily when the neuron of input layer 31 and corresponding neuron in output layer 33 are both ignite.

This is explained referring to Fig. 2 and Fig. 3. In this explanation, neuron Nij of output layer 33 is assumed to be the construct of a part of character code.

When neuron Nij in input layer 31 and neuron Nij in output layer 33 are both to be ignited for recognizing a

character, the weight distribution of synapses of middle layer 32 and output layer 33 so that neuron Nij in middle layer 32 ignite easily. For example, when neuron 1 in Fig. 3 is a neuron Nij of middle layer 32, the weight of the synapse corresponding to the input data from ignited neuron among the neurons of input layer connected to neuron 1 is increased by 5%. When input data "I2" and "I3" ignite, the weight of the synapse "W2" and "W3" are increased. The same processing is performed to the weight of synapse of neuron of output layer, that is, the weight of the synapse corresponding to the neuron to be ignited in middle layer is increased by 5%, for example.

From the third embodiment, the same effect of the first and the second embodiments can be obtained, too.

The third embodiment can be also applied t6o the neural network with 4 layers or more, other than the neural network with 3 layers, as the first and the second embodiments. In this case, selecting the first middle layer, the weight of the synapse of the middle layer and the output layer is increased so that each neuron in the middle layer obtains the tendency to ignite when the corresponding neurons in input layer and output layer are both ignite. When the learning up to the predetermined stage is completed, the second middle layer is added and the weight of synapse is increased in the second middle layer and the layer connected to the output side of the middle layer (output layer or the first middle layer) in the same way as the first middle layer. In this way, the weight distribution if obtained in the case of 4 layers. The same processing is executed in the case of more than 5 layers.

The fourth embodiment of the present invention is explained below.

It is considered that the ignition patterns of the layers 31, 32 and 33 change smoothly when the neuron in middle layer 32 has the tendency to ignite only in the case that at least one of corresponding neurons in input layer 31 and output layer 33 ignites. Therefore, in this embodiment, the weight of synapse of predetermined neuron is increased by predetermined rate in order that each neuron in middle layer 32 ignites easily when at least one of corresponding neurons in input layer 31 and output layer 33.

This is described in detail referring Fig. 2 and Fig. 3. A neuron Nij in output layer 33 is assumed to be a constituent part of a character code, as the case of the third embodiment.

When at least one of neurons Nij in input layer 31 and output layer 33 is to ignite for recognizing a character, the weight distribution of synapses in middle layer 32 and output layer 33 is to be decided in order for neuron Nij in middle layer 32 ignites easily. For example, when neuron 1 in Fig. 3 is the neuron Nij of middle layer 32, the weight of the synapse of input data from ignited neuron among the neurons in input layer connected to neuron 1 is increased by 5%. When input data is "I2" and "I3", the weight of the synapse to be increased weight is "W2" and "W3". The weight of the synapse of the neuron in output layer is performed the same processing. As described above, the weight of synapse of the neuron in middle layer to ignite is increased by 5%, for example.

From the fourth embodiment, the same effect of the embodiments above can be obtained.

The fourth embodiment can be also applied to the neural network with 4 layers or more, other than the neural network with 3 layers. In this case, selecting the first middle layer, the weight of the synapse of the middle layer and output layer is increased so that each neuron in the middle layer obtains the tendency to ignite when at least one neuron of the corresponding neurons in input layer and output layer ignites. When the learning up to the predetermined stage is completed, the second middle layer is added and the weight of synapse is increased in the second middle layer and the layer connected to the output side of the middle layer (output layer or the first middle layer) in the same way as the first middle layer. In this way, the weight distribution is obtained in the case of 4 layers. The same processing is executed in the case of more than 5 layers.

Input/output data for a neuron does not have to be the binary data. Multi-valued data and analog data are also acceptable.

Furthermore, an output layer is not necessary structured by the neurons on a diagonal line to express the character codes; it is possible to be structured by all neurons to define the recognition characters.

Number of neurons for each layer 31, 32 and 33 is not limited to 36 as the present embodiment. As many necessary neurons as the number of kinds of character to be recognized can be prepared.

It is possible to apply the present invention not only to the character recognition but also to the configuration recognition or to acoustic recognition.

According to the present invention as mentioned above, it is possible to obtain an effect that an efficient learning can be performed onto the middle layer of the neural network.

Claims

1. A training method for transforming an untrained network into a trained network, said network comprising an input layer (31), a middle layer (32) and an output layer (33), each said layer including a plurality of neurons, each said neuron being a signal processing element, said middle layer being arranged between said input and output layers said method being characterised in that each neuron in said middle layer is connected to each neuron in said input and output layers, and by the steps of:

   determining an activation pattern of said neurons in said input layer;

determining an activation pattern of said neurons in said output layer;

for each neuron of the middle layer increasing the weight by which connections between neurons of the input layer and said neurons in the middle layer and increasing the weights corresponding to connections between said neuron of the middle layer and neurons of the output layer if this increasing contributes to a better approximation of the ignition pattern of the middle layer to the ignition patterns of the input layer and the output layer; and

repeating the increasing step for each neuron of the middle layer, thereby generating the nearest

approximation of the ignition pattern of the middle layer to the ignition patterns of the input layer and the output layer.

2. A training method according to claim 1, further comprising the steps of:
adding an additional middle layer between said input layer and said middle layer or between said middle layer and said output layer and performing the increasing step for the additional middle layer in the same manner as for the first middle layer.

3. A training method according to claim 1 or 2, wherein the increasing of the weights is performed in order to minimize the sums of corresponding absolute differences between the output values of neurons of the input layer and the middle layer and between the output values of the middle layer and the output layer.

4. A training method according to claim 1 or 2, wherein the increasing of the weights for a certain neuron of the middle layer is performed when the corresponding neurons in the input and output layers both ignite.

5. A training method according to claims 1 or 2, wherein the increasing of the weights for a particular neuron of the middle layer is performed when either of the corresponding neurons of the input layer and the output layer ignites.

Patentansprüche

1. Trainingsverfahren zum Umwandeln eines untrainierten Netzwerkes in ein trainiertes Netzwerk, wobei das Netzwerk eine Eingangsschicht (31), eine Mittelschicht (32) und eine Ausgangsschicht (33) aufweist, wobei jede Schicht eine Vielzahl von Neuronen aufweist und jedes Neuron ein Signalverarbeitungselement darstellt, wobei die mittlere Schicht zwischen der Eingangs- und Ausgangsschicht angeordnet ist, und wobei das Verfahren dadurch gekennzeichnet ist, daß jedes Neuron in der mittleren Schicht mit jedem Neuron in der Eingangs- und Ausgangsschicht verbunden ist sowie durch folgende Schritte:

Bestimmen eines Zündungsmusters der Neuronen in der Eingangsschicht;

Bestimmen eines Zündungsmusters der Neuronen in der Ausgangs schicht;

für jedes Neuron der mittleren Schicht Erhöhen der Gewichte der Verbindungen zwischen Neuronen der Eingangsschicht und der Neuronen in der mittleren Schicht und Erhöhen der Gewichte der Verbindungen zwischen dem Neuron der mittleren Schicht und den Neuronen der Ausgangsschicht, falls die Erhöhung zu einer besseren Annäherung des Zündmusters der mittleren Schicht zu den Zündmustern der Eingangsschicht und der Ausgangsschicht führt, und

Wiederholen des Erhöhungsschrittes für jedes Neuron der mittleren Schicht, wodurch die beste Aproximation des Zündungsmusters der mittleren Schicht zu den Zündungsmustern der Eingangsschicht und der Ausgangsschicht erzeugt wird.

2. Trainingsverfahren nach Anspruch 1, weiterhin aufweisend die Schritte:
Addieren einer zusätzlichen mittleren Schicht zwischen der Eingangsschicht und der mittleren Schicht oder zwischen der mittleren Schicht und der Ausgangsschicht und Ausführen des Erhöhungsschrittes für die zusätzliche mittlere Schicht in der gleichen Weise wie für die erste mittlere Schicht.

3. Trainingsverfahren nach Anspruch 1 oder 2, wobei das Erhöhen der Gewichte ausgeführt wird, um die Summe der entsprechenden absoluten Differenzen zwischen den Ausgangswerten der Neuronen der Eingangsschicht und der mittleren Schicht und zwischen den Ausgangswerten der mittleren Schicht und der Ausgangsschicht zu minimieren.

4. Traningsverfahren nach Anspruch 1 oder 2, wobei das Erhöhen der Gewichte für ein bestimmtes Neuron der mittleren Schicht ausgeführt wird, wenn die entsprechenden Neuronen in der Eingangs- und Ausgangsschicht beide zünden.

5. Trainingsverfahren nach Anspruch 1 oder 2, wobei

das Erhöhen der Gewichte für ein bestimmtes Neuron der mittleren Schicht ausgeführt wird, wenn entweder die entsprechenden Neuronen der Eingangsschicht oder der Ausgangsschicht zünden.

Revendications

1. Procédé d'apprentissage pour transformer un réseau non formé en un réseau formé, ledit réseau comprenant une couche d'entrée (31), une couche médiane (32) et une couche de sortie (33), chacune desdites couches contenant une pluralité de neurones, chaque neurone étant un élément de traitement de signaux, ladite couche médiane étant placée entre lesdites couches d'entrée et de sortie, ledit procédé étant caractérisé en ce que chaque neurone de ladite couche médiane est connecté à chaque neurone desdites couches d'entrée et de sortie et par les étapes consistant à :

    déterminer une configuration de l'activation desdits neurones de ladite couche d'entrée ;
    déterminer une configuration de l'activation desdits neurones de ladite couche de sortie ;
    pour chaque neurone de la couche médiane, augmenter les poids des connexions entre des neurones de la couche d'entrée et lesdits neurones de la couche médiane et augmenter les poids correspondants des connexions entre ledit neurone de la couche médiane et des neurones de la couche de sortie si cette augmentation contribue à une meilleure approximation entre le diagramme d'excitation de la couche médiane et les configurations d'excitation de la couche d'entrée et de la couche de sortie ; et
    répéter l'étape d'augmentation pour chaque neurone de la couche médiane en créant ainsi l'approximation la plus proche de la configuration d'excitation de la couche médiane par rapport aux configurations d'excitation de la couche d'entrée et de la couche de sortie.

2. Procédé d'apprentissage selon la revendication 1, comprenant encore les étapes consistant à :
    ajouter une couche supplémentaire médiane entre ladite couche d'entrée et ladite couche médiane ou entre ladite couche médiane et ladite couche de sortie et effectuer l'étape d'augmentation pour la couche médiane supplémentaire de la même manière que pour la première couche médiane.

3. Procédé d'apprentissage selon la revendication 1 ou 2, dans lequel l'augmentation des poids est effectuée dans le but de minimiser les sommes des différences absolues correspondantes entre les valeurs de sortie des neurones de la couche d'entrée et de la couche médiane et entre les valeurs de sortie de la couche médiane et de la couche de sortie.

4. Procédé d'apprentissage selon la revendication 1 ou 2, dans lequel l'augmentation des poids pour un certain neurone de la couche médiane est effectuée quand les neurones correspondants de la couche d'entrée et de la couche de sortie sont tous deux excités.

5. Procédé d'apprentissage selon la revendication 1 ou 2, dans lequel l'augmentation des poids pour un neurone particulier de la couche médiane est effectuée quand l'un ou l'autre des neurones correspondants de la couche d'entrée et de la couche de sortie est excité.

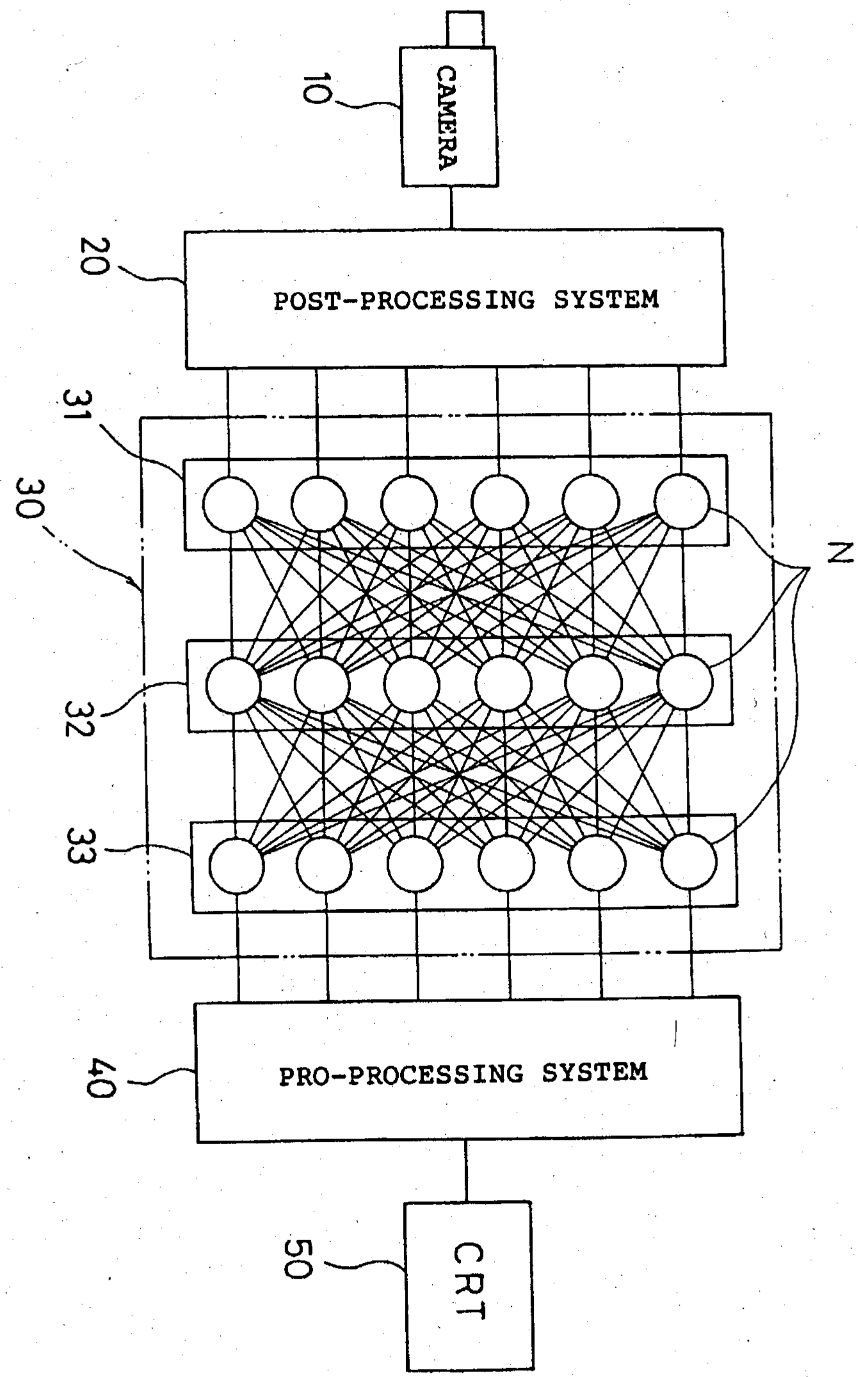
10
CAMERA
20
POST-PROCESSING SYSTEM
31
30
32
33
N
40
PRO-PROCESSING SYSTEM
50
CRT

FIG. 1

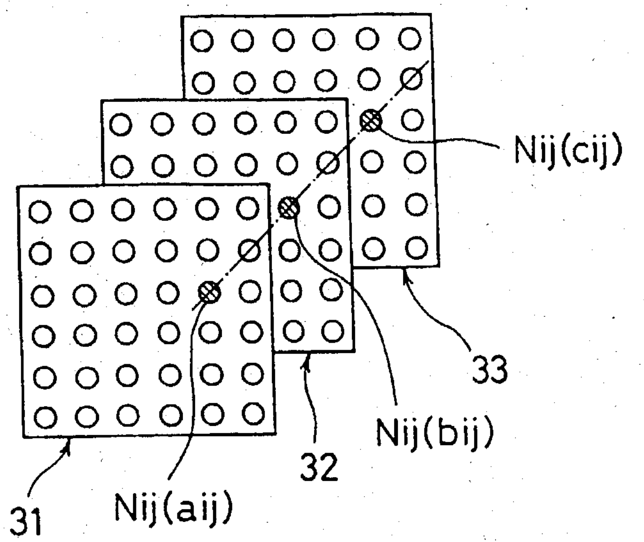

FIG. 2

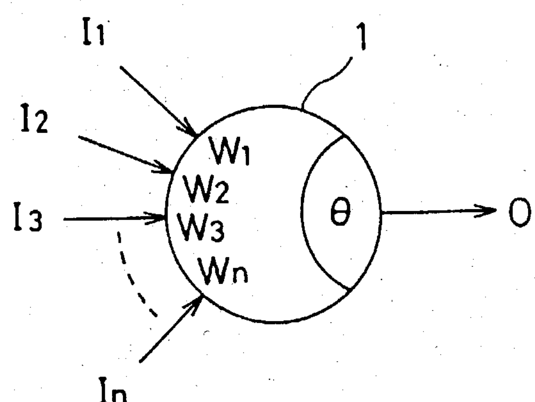

FIG. 3

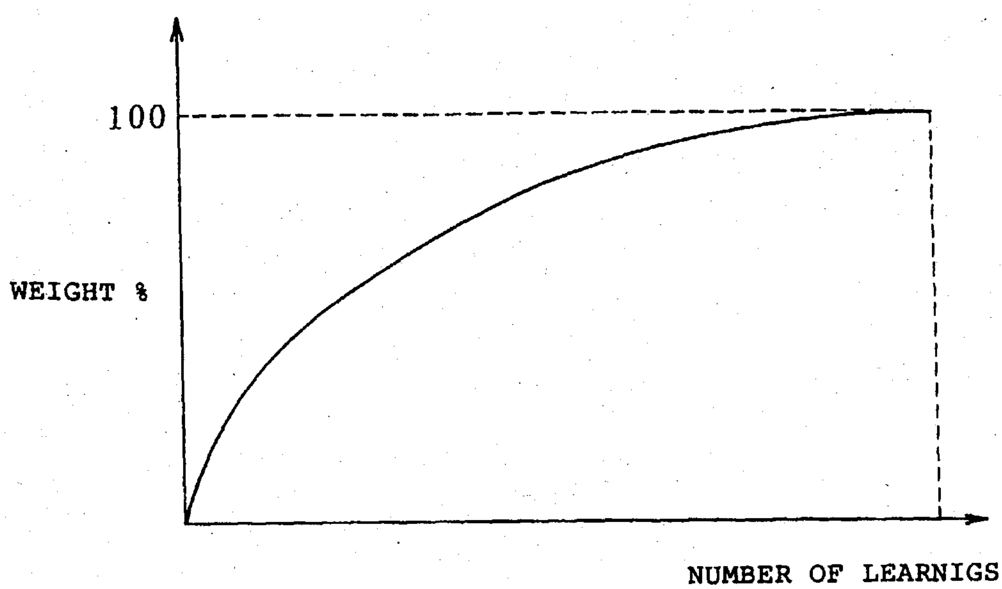

FIG. 4